CLAIMS

1. A method for tunneling data associated with a packet based multimedia communication standard, comprising:

intercepting a library call associated with the multimedia communication standard;

registering identification data associated with the library call;

adding a Transmission Control Protocol/Internet Protocol (TCP/IP) header over a pre-existing header of a data packet related to the identification data; and

transmitting the data packet having the (TCP/IP) header through a firewall.

2. The method of claim 1, wherein the method operation of registering identification data associated with the library call includes,

checking if a port number and a protocol type are defined in a table;

if the port number and the protocol type are not defined in the table, the method includes,

adding the port number and the protocol type to the table.

3. The method of claim 1, wherein the identification data includes a port number and a protocol type associated with the data packet.

4. The method of claim 1, wherein the method operation of adding a Transmission Control Protocol/Internet Protocol (TCP/IP) header over a pre-existing header of a data packet related to the identification data includes,

inserting a flag into a lower byte of a window size field of the TCP/IP header; and

inserting a checksum into an upper byte of the window size field of the TCP/IP header.

5. The method of claim 2, wherein the port number is port 80 and the protocol type is a TCP.

6. The method of claim 1, wherein the method operation of registering identification data associated with the library call is done prior to advancing data associated with the library call from an application level of a protocol stack of the packet based multimedia communication standard to a driver level of the packet based multimedia communication standard.

7. A method for communicating port traffic through a single Hypertext Transfer Protocol (HTTP) port, comprising:

a) establishing a connection between a first and second computing device;

b) transmitting allocation data associated with the port traffic to a tunneling driver;

c) segmenting the port traffic into datagrams;

d) appending a first header to each one of the datagrams; and

e) appending a Transmission Control Protocol/ Internet Protocol (TCP/IP) header over the first header, wherein the TCP/IP header is configured to direct each one of the datagrams to the single HTTP port.

8. The method of claim 7, wherein the connection is a TCP connection and the single HTTP port is port 80.

9. The method of claim 7, wherein the method operation of transmitting allocation data includes,

defining a port number and a protocol type associated with the port traffic.

10. The method of claim 7, further comprising:

setting a SYN flag in the TCP/IP header for initiation of the connection from behind a firewall; and

setting SYN+ACK flags in the TCP/IP header for responses to the initiation of the connection from outside of the firewall.

11. The method of claim 7, wherein the method operation of appending a Transmission Control Protocol/ Internet Protocol (TCP/IP) header over the first header includes,

inserting a flag into a lower byte of a window size field of the TCP/IP header; and

inserting a checksum into an upper byte of the window size field of the TCP/IP header.

12. A computer readable medium having program instructions for tunneling data associated with a packet based multimedia communication standard, comprising:

program instructions for intercepting a library call associated with the multimedia communication standard;

program instructions for registering identification data associated with the library call;

program instructions for adding a Transmission Control Protocol/Internet Protocol (TCP/IP) header over a pre-existing header of a data packet related to the identification data; and

program instructions for transmitting the data packet having the (TCP/IP) header through a firewall.

13. The computer readable medium of claim 12, wherein the program instructions for operation of registering identification data associated with the library call includes,

program instructions for checking if a port number and a protocol type are defined in a table; and

program instructions for adding the port number and the protocol type to the table.

14. The computer readable medium of claim 12, wherein the identification data includes a port number and a protocol type associated with the data packet.

15. The computer readable medium of claim 12, wherein the program instructions for adding a Transmission Control Protocol/Internet Protocol (TCP/IP)

header over a pre-existing header of a data packet related to the identification data includes,

program instructions for inserting a flag into a lower byte of a window size field of the TCP/IP header; and

program instructions for inserting a checksum into an upper byte of the window size field of the TCP/IP header.

16. The computer readable medium of claim 13, wherein the port number is port 80 and the protocol type is a TCP.

17. The computer readable medium of claim 12, wherein the program instructions for registering identification data associated with the library call are completed prior to advancing data associated with the library call from an application level of a protocol stack of the packet based multimedia communication standard to a driver level of the packet based multimedia communication standard.

18. A computer readable medium having program instructions for communicating port traffic through a single Hypertext Transfer Protocol (HTTP) port, comprising:

a) program instructions for establishing a connection between a first and second computing device;

b) program instructions for transmitting allocation data associated with the port traffic to a tunneling driver;

c) program instructions for segmenting the port traffic into datagrams;

d) program instructions for appending a first header to each one of the datagrams; and

e) program instructions for appending a Transmission Control Protocol/ Internet Protocol (TCP/IP) header over the first header, wherein the TCP/IP header is configured to direct each one of the datagrams to the single HTTP port.

19. The computer readable medium of claim 18, wherein the connection is a TCP connection and the single HTTP port is port 80.

20. The computer readable medium of claim 18, wherein the program instructions for transmitting allocation data includes,

program instructions for defining a port number and a protocol type associated with the port traffic.

21. The computer readable medium of claim 18, wherein the first header is one of a TCP header and a User Datagram Protocol (UDP).

22. The computer readable medium of claim 18, wherein the program instructions for appending a Transmission Control Protocol/ Internet Protocol (TCP/IP) header over the first header includes,

program instructions for inserting a flag into a lower byte of a window size field of the TCP/IP header; and

program instructions for inserting a checksum into an upper byte of the window size field of the TCP/IP header.

23. A system for tunneling port traffic destined for multiple ports through a single port, comprising:

a server configured to transmit data packets each having a tunneling header in addition to a packet header;

a firewall limiting a number of unblocked TCP ports, the firewall capable of analyzing the tunneling header, wherein the tunneling header is associated with the single port so that the firewall allows the data packets to pass through; and

a client configured to receive the data packets from the firewall through the single port, the client further configured to identify a flag and a checksum associated with the tunneling header in order to strip the tunneling header for access to the packet header.

24. The system of claim 23, wherein the flag and the checksum are incorporated into a window size region of the tunneling header.

25. The system of claim 23, wherein the packet header is a User Datagram Protocol (UDP) packet header.

26. The system of claim 23, wherein the single port is hypertext transfer protocol (HTTP) port 80.

27. A communication protocol stack for enabling multimedia communication between communicating devices, comprising:

at an application level, identifying whether received communication data is for a communication port; and

if the received communication data is for the communication port, forwarding identification data regarding the received communication data to a table in advance of forwarding the received communication data to a driver level of the communication protocol stack.

28. The communication protocol stack of claim 27, further comprising:

forwarding the communication data to the driver level; and

checking whether the identification data associated with the communication data is listed in the table.

29. The communication protocol stack of claim 28, further comprising:

if the identification data associated with the communication data is listed in the table, inserting a tunneling header over a header of the communication data; and

incorporating a firewall Internet protocol address and a firewall port number into the tunneling header.

30. The communication protocol stack of claim 27, wherein a tunneling driver associated with the driver level inserts a tunneling header over a header of the communication data when the communication data is for the communication port.